Mesh-based tools for land ice simulations

Tim Tautges

Dmitry Karpeev

Argonne National Laboratory

CCSM Land Ice Working Group Summer Meeting

Breckenridge, CO

June 30, 2010

Outline

- Sisiphus overview
- Mesh-based geometry
- Solver issues

Scalable Ice-sheet Solvers and Infrastructure for Petascale, High-resolution, Unstructured Simulations

Timothy J. Tautges (PI), Barry Smith, Dmitry Karpeev, Jean Utke (ANL)

Jed Brown (ETH-Zurich), Patrick Heimbach (MIT) Bill Lipscomb (LANL)

Non-Newtonian Stokes system:

$$-\nabla \bullet (\eta D\boldsymbol{u}) + \nabla p - \boldsymbol{f} = 0$$
$$\nabla \bullet \boldsymbol{u} = 0$$

with boundary conditions for:

$$(Du - p1) \bullet n = \begin{cases} 0 & \textbf{free surface} \\ -\rho_w z n & \textbf{ice-ocean} \end{cases}$$

$$u = 0 \quad \textbf{frozen bed}$$

$$\left. \begin{aligned} u \bullet n &= g_{melt}(Tu, \ldots) \\ T(Du - p1) \bullet n &= g_{slip}(Tu, \ldots) \end{aligned} \right\} \quad \textbf{non-linear slip}$$

Navier, Weerman, or Coulomb power law for g_{slip}

Mesh motion:

$$-\nabla \bullet \sigma = 0$$

$$\sigma = \mu\left[2Dw + (\nabla w)^T \nabla w\right] + \lambda tr(\nabla w)1, \quad w = x - x_o$$

$$\text{surface}: (\dot{x} - u) \bullet n = q_{BL}, T\sigma \bullet n = 0$$

Enthalpy transport:

$$\rho\left[\frac{\partial}{\partial t}\Theta + \underbrace{(u - \dot{x}) \bullet \nabla\Theta}_{\text{ALE}}\right] - \nabla \bullet \left[\underbrace{\kappa(\Theta)\nabla\Theta}_{\text{Fourier/Fick diffusion}} + \underbrace{q_D(\Theta)}_{\text{Darcy flow}}\right] - \underbrace{\eta Du : Du}_{\text{Strain heating}} = 0$$

Modeling:

- Hp-adaptive FEM in space, fully implicit in time

Preconditioning:

- "Dual-order" over space – high-order FEM, preconditioned with low-order linear elements from high-order nodes
- Block-ILU, replacing sub-blocks with physics-based equivalents

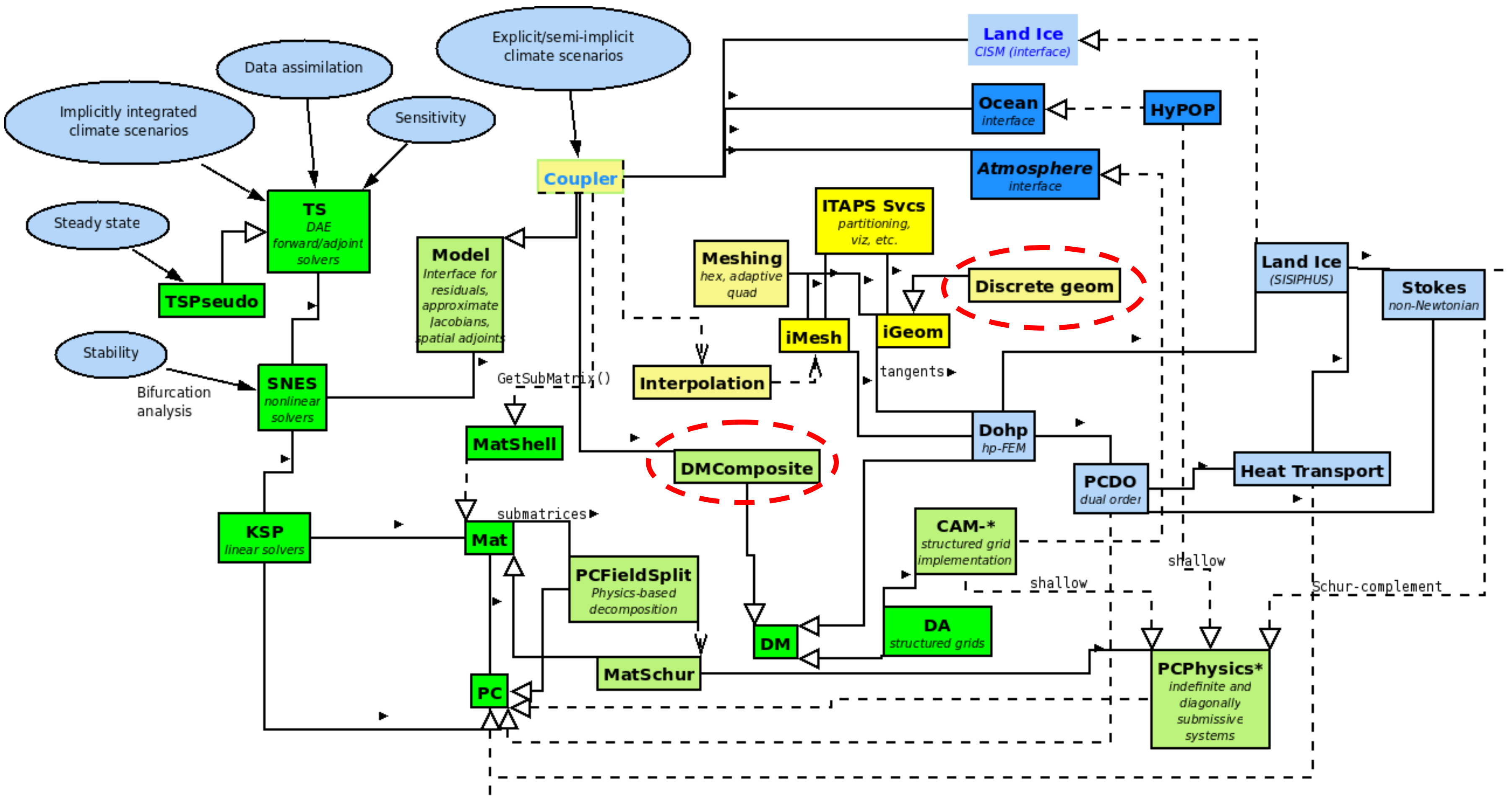
Geometry/mesh:

- Unstructured, hexahedral extruded mesh
- Mesh-based geometry w/ smooth normals for bed, ice surface
- Adaptive mesh near bed, grounding line

Implementation:

- Use component-based solvers (PETSc), tools (ITAPS)
- Higher-level interface to Petsc for expressing physics and physics-based preconditioners
- Use Petsc Data Manager (DM) implementation based on ITAPS mesh interface

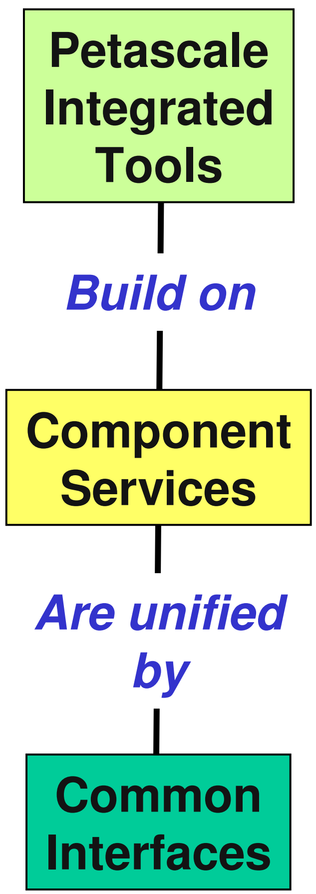
SISIPHUS Software Component View



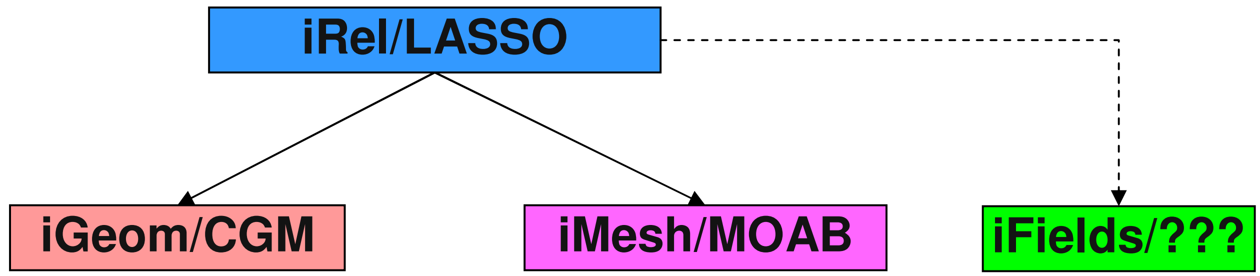
PETSc | MOAB/CGM | ice physics

ITAPS In One Slide

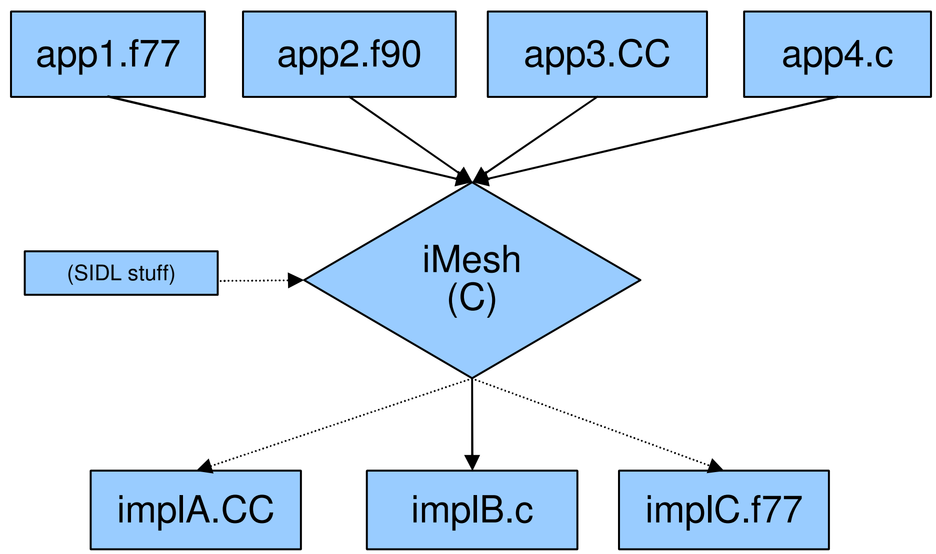
37k foot view:



Interface relationships:



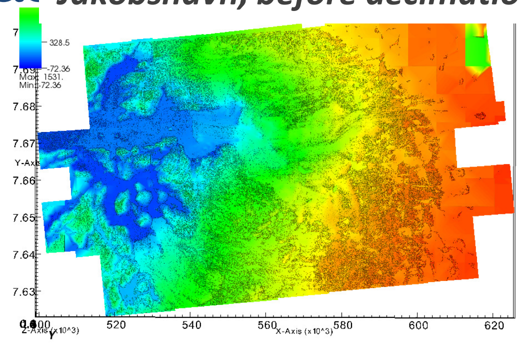
Application view:



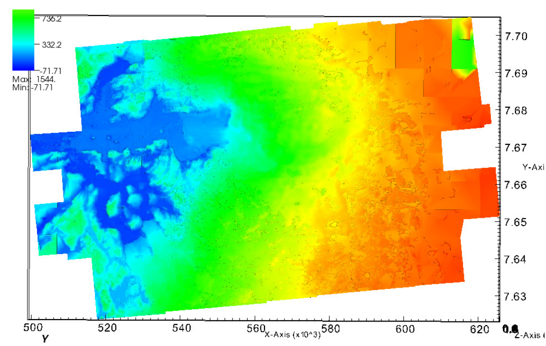
Task #1: Representation of ice sheet bed, surface a

- 2 primary sources of data:
  - CReSIS flight path data (<< 5km)
  - ISIS (J. Johnson, UMT) data sets (5km)

Jakobshavn, before decimation (5M tri)



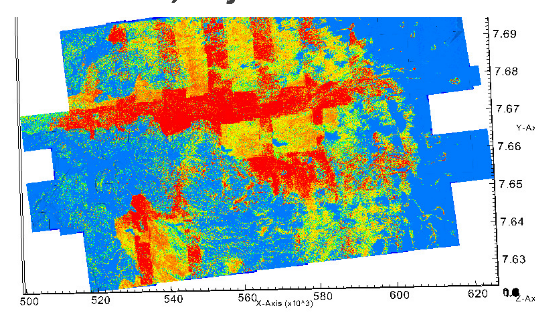
Jakobshavn, after decimation (200k tri)



- CRESIS data
  - Read as points, elevations
  - Triangulate using Triangle
  - Decimate

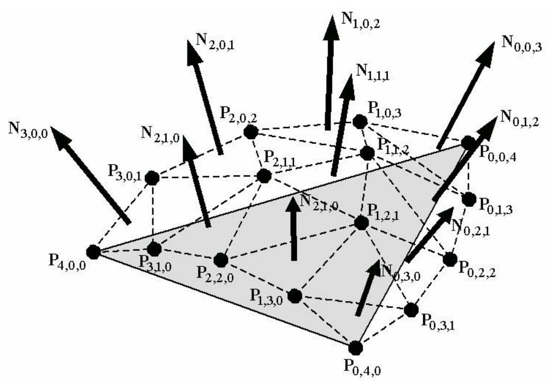
- Decimation
  - Using Qslim algorithm (Garland & Heckbert, Siggraph ’97)
  - Implemented on MOAB
  - Challenge: noisy data, reasonable run-times

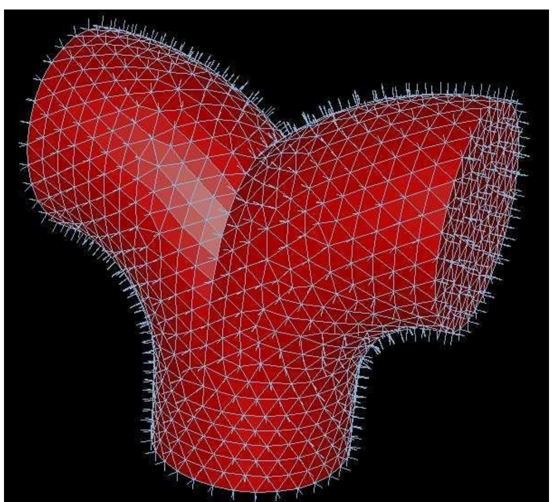
Thickness, before decimation

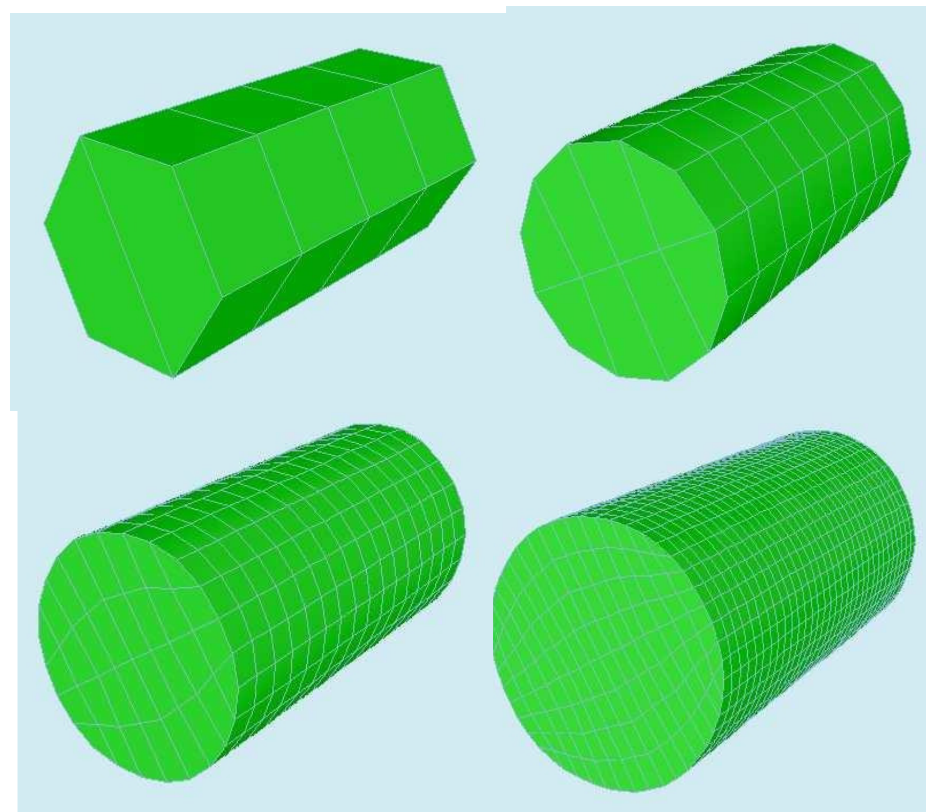


Task #2: Smooth tangents, normals on facet-based surface

- C1-continuous facet-based geometric representation to support meshing
  - Owen, White, Tautges, “Facet-based surfaces for 3d mesh generation”, 11^{th} IMR, ’02)

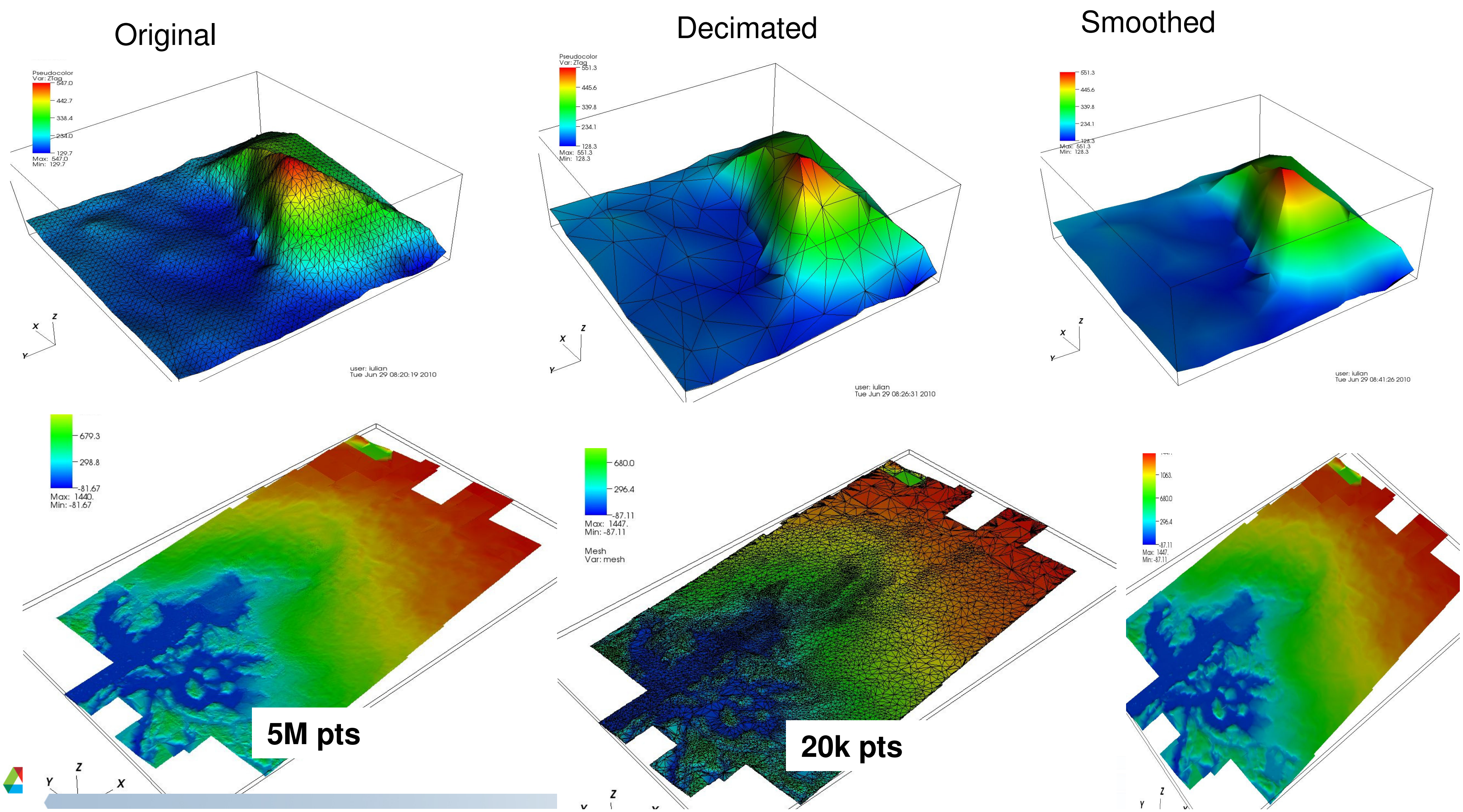






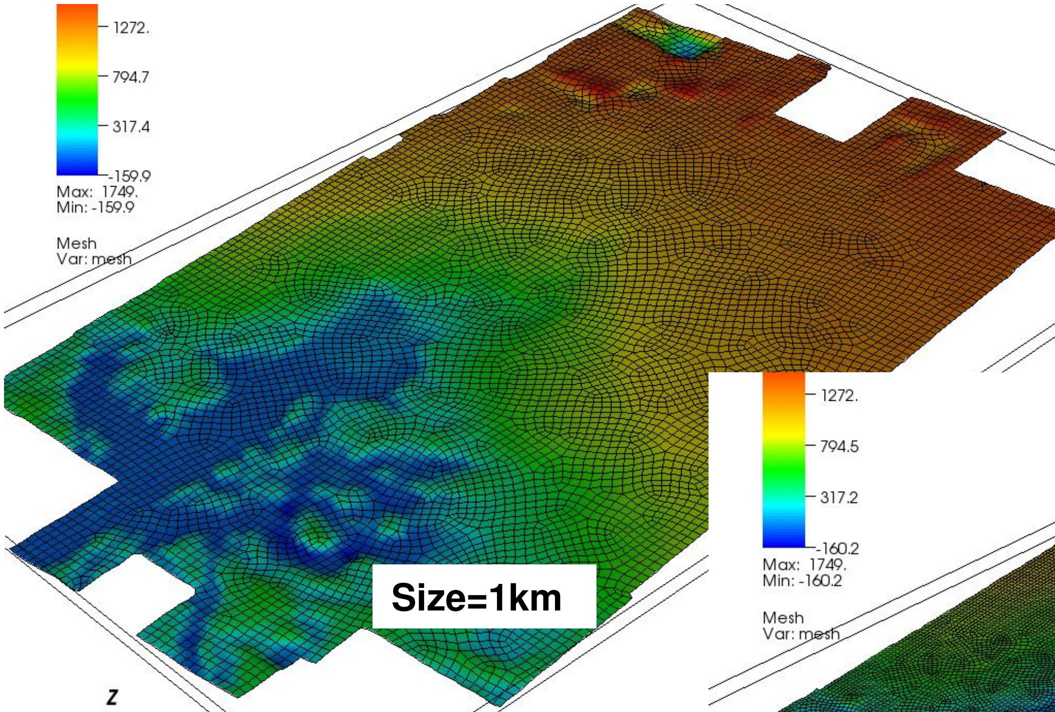
Task #2: Smooth tangents, normals on facet-based surface

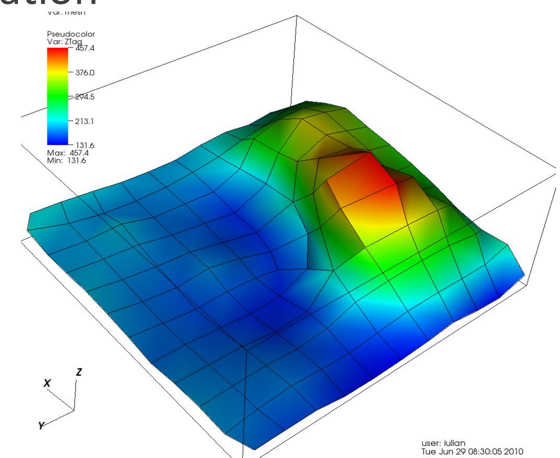
- For ice sheet data…

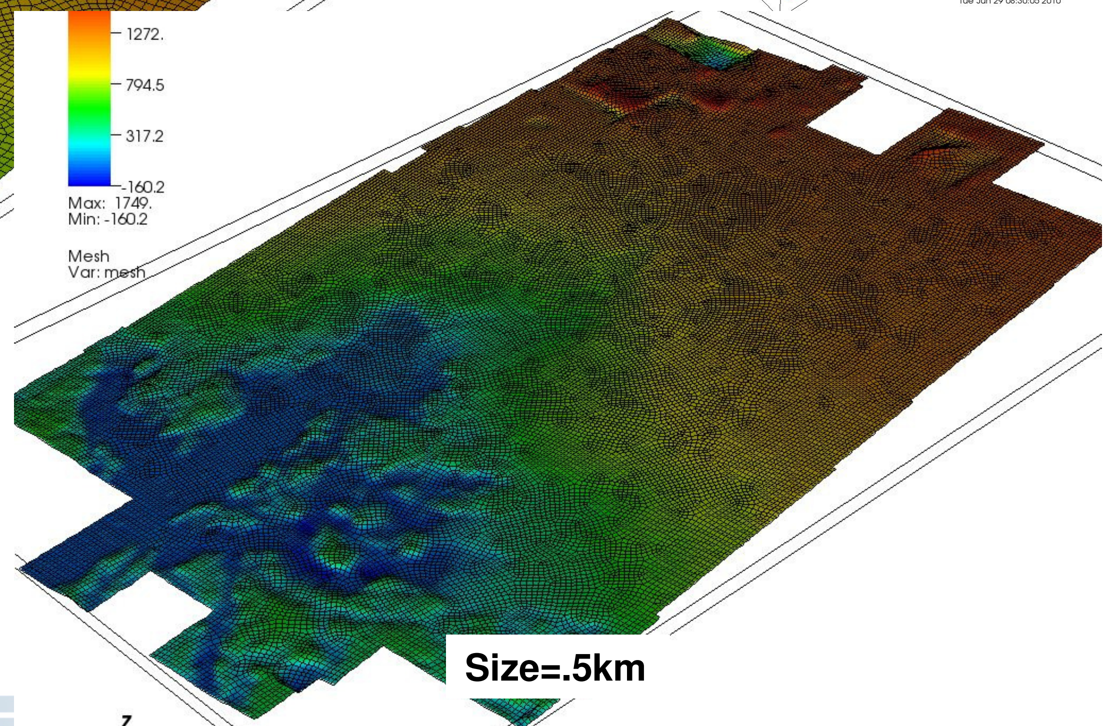


Task #3: Quad meshing on smoothed surfaces

- Present an iGeom interface for smoothed surface representation
  - Can feed directly to quadrilateral mesh generator







- Other mesh algs also apply
- Need to improve eval speed

Solvers: simplify interaction with mesh MatDD: new PETSc matrix interface

$$M = \sum_{ij} G_i M_{ij} S_j, \quad S_j : V \rightarrow \Re^{n_j}, \quad G_i : \Re^{m_i} \rightarrow U$$

$$u \xleftarrow{G} \begin{pmatrix} u_1 \\ \vdots \\ u_i \end{pmatrix} \begin{pmatrix} M_{11} & \cdots & M_{1j} \\ \vdots & \cdots & \vdots \\ M_{i1} & \cdots & Mij \end{pmatrix} \begin{pmatrix} v_1 \\ \vdots \\ v_j \end{pmatrix} \xleftarrow{S} v$$

- Blocks $\{M_{ij}\}$ applied, preconditioned, inverted separately
- Block structure can be used recursively, subsystems assembled
- Gather $\{G_i\}$ and scatter $\{S_j\}$ encode space splitting
- Enable both factorization (splitting of assembled matrices) and DD (assembly out of blocks) PCs
- Global scatter/gather translates loosely to local/non-local mesh
- iField: local formulation of operators (gradient, integral, etc.) on elements based on local dof arrays

Conclusions

- Moving toward mesh-based representation of ice sheet geometry, read directly from CRESIS or other .nc-based data
- Represented in a form which directly supports mesh generation and geometric (tangent, normal) queries
- Incorporating higher-level support in PETSc for expressing factorization- and DD-based preconditioners